IN THE SPECIFICATION

Please replace the paragraph at page 22, prenumbered lines 1-8, with the following rewritten paragraph:

By the reading part 141, T samples x(L-T-1) to x(L-2) preceding the last sample x(L-1) are read out of the buffer 100 as the sample sequence ΔS' consisting of consecutive samples forming part of the current frame, then the sample sequence ~~AS~~ ΔS' is rearranged in a reverse order in a reverse arrangement part 142, from which the samples x(L-2), x(L-3), ..., x(L-T-1) are provided as an alternative sample sequence AS', and the alternative sample sequence AS' is stored by the writing part 143 in the buffer 100 so that it is concatenated to the last sample x(L-1) of the current frame.

Please replace the paragraph at page 27, prenumbered lines 18-27, with the following rewritten paragraph:

These predetermined generating methods are prestored in a generation method storage part 160 in Fig. 9A, and under the control of a select control part 170, one of the alternative sample sequence generating method is read out from the generation method storage part 170 and set in an alternative sample sequence generating part 110; the alternative sample sequence generating part 110 begins to operate, and follows the generating method set therein to take out of the buffer 100 a sample sequence ΔS, which consists of consecutive samples forming part of the current frame, and to generate an alternative sample sequence (a candidate), which is provided to the select control part 170.

Please replace the paragraph at page 28, prenumbered lines 13-21, with the following rewritten paragraph:

If the input candidate alternative sample <u>sequence</u> is the sample sequence AS corresponding to that of the preceding frame, it is stored in a register 174, and the square error between the sample sequence AS and the sample sequence x(-T), ..., x(-1) stored in the register 172 is calculated in a distortion calculating part 175. If the input candidate alternative sample <u>sequence</u> is the sample sequence AS' corresponding to that of the succeeding frame, it is stored in a register 176, and the square error between the sample sequence AS' and the sample sequence x(L), ..., x(L+T-1) stored in the register 173 is calculated in the distortion calculating part 175.

Please replace the paragraph at page 33, prenumbered lines 10-13, with the following rewritten paragraph:

The sample sequence v(-p), ..., v(L-1) with the alternative sample concatenated thereto is input to the prediction error generating part [[5]] <u>51</u>, which generates a prediction error signal y(0), ..., y(L-1) by autoregressive prediction (S5).

Please replace the paragraph at page 33, prenumbered line 14, to page 34, prenumbered line 11, with the following rewritten paragraph:

The position τ of the similar sample sequence x(n+ τ), ..., x(n+τ+p-1) and the gain β are determined such that, for example, the power of the prediction error signal y(0), ..., y(L-1) becomes minimum. In this instance, τ and β are determined using the power of the prediction error signal from y(0) to y(2p) because once the calculation of the prediction value comes to use p samples subsequent to v(p) the prediction error power is not related to the part in the in the current frame from where the similar sample sequence x(n+τ), ..., x(n+τ+p-1) is

derived. The method of this determination is the same as the alternative sample sequence AS determining method described previously with reference to Fig. 10. In this case, upon each change of τ the error power is calculated in an error power calculating part 250 (Fig. 11), and when the calculated value is smaller than the minimum value p_{EM} obtained until then, the latter is updated with the newly calculated value, which is stored as the minimum value p_{EM} in a memory ~~260~~ 265, and the similar sample sequence obtained at that time is also stored in the memory ~~260~~ 265, updating the previous sequence stored therein. Then τ is changed to the next τ, that is, $\tau \leftarrow \tau+1$, and the error power is calculated, and if the error power is not smaller than the previous one, the similar sample sequence at that time is stored in the memory ~~26~~ 265, updating the previous sample sequence stored therein; the similar sample sequence stored at the time of completion of changing τ from 1 to L-1-p is adopted. Next, β is changed on a stepwise basis for the adopted similar sample sequence; each time it is change, the error power is calculated, and β is adopted corresponding to the minimum power of prediction error. The determination of τ and β is made under the control of the selection/determination control part 260 (Fig. 11).

Please replace the paragraph at page 36, prenumbered lines 10-22, with the following rewritten paragraph:

The auxiliary information decoding part 330 decodes the auxiliary code C_{AI} forming part of the code of the current frame FC to obtain auxiliary information, from which τ and β are obtained (S4). The auxiliary information decoding part ~~320~~ 330 may sometimes be supplied with the auxiliary information itself. In a sample sequence acquiring part 340, τ is used to replicate from the synthesis signal (sample) sequence a sample sequence $v(\tau)$, ..., $v(\tau+p)$ consisting of a predetermined number p of consecutive samples in this case, that is, the prediction synthesis signal sequence v(n) is obtained intact as the replicated sample

sequence $v(\tau)$, …, $v(\tau+p)$ (S5), then this sample sequence is so shifted as to bring its forefront to the front position of the frame FC to provide the sample sequence u(n), which is multiplied by the gain β from the auxiliary information in a gain multiplying part 350 to generate a corrected sample sequence $u(n)'=\beta u(n)$ (S6).

Please replace the paragraph at page 38, prenumbered lines 16-20, with the following rewritten paragraph:

~~Thereafter~~ Similar prediction (prediction with progressive order) is continued. Namely, upon each input of a sample a convolution is carried out between a prediction coefficient of the prediction order increased one by one ~~through use of all preceding samples until then~~ and the preceding samples to obtain a prediction value, and the prediction value is subtracted from the input sample at that time to obtain a prediction error signal.

Please replace the paragraph at page 39, prenumbered line 13, to page 40, prenumbered line 6, with the following rewritten paragraph:

An example of the procedure described above is shown in Fig. 18. In the first place, n is initialized to 0 (S1), then the sample x(0) is rendered into the prediction error signal y(0) (S2), then n is incremented by one (S3), then the nth-order prediction coefficients $\alpha^{(n)}{}_1$, …, ~~$\alpha(n)$~~ $\underline{\alpha^{(n)}{}_n}$ are calculated ~~from the past samples x(0), …, x(n-1)~~ (S4), then the past samples x(0), …, x(n-1) are convoluted with the prediction coefficients to obtain prediction values, then the prediction values are each subtracted from the input current sample x(n) to obtain the prediction error signal y(n) (S5). That is, the following calculation is conducted.

$$y(n) = x(n) - \sum_{i-1}^{n} \alpha_i^{(n)} x(n-i)$$

A check is made to see if n is p (S6), and if not, then the procedure returns to step S3, and if n=p, then the pth-order prediction coefficients $\alpha^{(p)}_1, \ldots, \alpha^{(p)}_p$ are calculated from all the samples x(0), …, x(L-1) (S7), then a convolution is carried out between the prediction coefficients and the immediately preceding p past samples x(n-p), …, x(n-1) to obtain a prediction value, and the prediction value is subtracted from the current sample x(n) to obtain the prediction error signal y(n) (S8). In other words, Eq. (2) is calculated. A check is made to see if processing of all required samples is completed (S9), and if not, then n is incremented by one and the procedure returns to step S8 (S10); if completed, the processing ends.

Please replace the paragraph at page 40, prenumbered lines 7-15, with the following rewritten paragraph:

Fig. 19 presents in tabular form the prediction coefficients $\alpha^{(n)}_1, \ldots, \cancel{\alpha^{(n)}_p}\ \underline{\alpha^{(n)}_n}$ that are generated for each sample number n=0, …, L-1 of the current frame in the case of applying Embodiment 6 to the prediction error generation in Fig. 3A. No prediction is made for the sample x(0) of the first sample number n=0 of the current frame. For the respective samples x(n) of the next sample number n=1 to n=p-1, the nth-order prediction coefficients $\alpha^{(n)}_1, \ldots, \alpha^{(n)}_n$ are sets, and the remaining (p-n) coefficients are set to $\cancel{\alpha^{(n)}_{n+2}}\ \underline{\alpha^{(n)}_{n+1}} = \cancel{\alpha^{(n)}_{n+3}}\ \underline{\alpha^{(n)}_{n+2}}, \ldots = \alpha^{(n)}_p = 0$. For each sample x(n), where n=p, …, L-1, the pth-order prediction coefficients $\alpha^{(p)}_1, \ldots, \alpha^{(p)}_p$ are calculated and set.

Please replace the paragraph at page 40, prenumbered lines 16-24, with the following rewritten paragraph:

Since the pth-order linear prediction requires past p samples, the prediction for the leading samples x(0), …, x(p-1) of the current frame calls for rear-end samples of the

preceding frame, but as in Embodiment 6, by sequentially increasing the prediction order progressively from 0 to p-1(progressive order) for the samples of sample numbers n=0 to n=p-1 and by performing the pth-order prediction for the samples after the sample number n=p, (consequently, by performing the prediction without using samples of the preceding frame), it is possible to reduce discontinuity of the prediction signal between the preceding and current frames.

Please replace the paragraph at page 40, prenumbered line 26, to page 41, prenumbered line 10, with the following rewritten paragraph:

Fig. 20 illustrates Embodiment 7 of the prediction synthesis processing (applied to Embodiment [[4]] 6 of Fig. 4A) corresponding to Fig. 17. A prediction coefficient decoding part [[66]] 66D decodes pth-order prediction coefficients from its received auxiliary information, and calculates nth-prediction coefficients (n=1, …, p-1) from the pth-prediction coefficients. Upon input of the first one y(0) of the prediction error signals y(0), …, y(L-1) of the current frame FC, it is out put intact as a prediction synthesis signal x(0). Upon input of the next prediction error signal y(1), a convolution, $\cancel{\alpha^{(1)}_1 y(0)}$ $\alpha^{(1)}_1 x(0)$, is conducted in the multiplying part M_1 between the 1st-order prediction coefficient $\alpha^{(1)}_1$ obtained from the prediction coefficient decoding part [[66]] 66D and the ~~prediction error signal y(0)~~ x(0) to obtain a prediction value, which is added to y(1) to obtain a synthesis signal x(1).

Please replace the paragraph at page 41, prenumbered line 11, to page 42, prenumbered line 7, with the following rewritten paragraph:

Upon input of the next prediction error signal y(2), a convolution is conducted in the multiplying part M_2 between the 2nd-order prediction coefficients $\alpha^{(2)}_1$, $\alpha^{(2)}_2$ from the prediction coefficient decoding part [[66]] 66D and ~~the prediction error signal y(0), y(1)~~ x(0),

x(1) to obtain a prediction value, which is added to y(2) to obtain a synthesis signal x(2). Thereafter, upon input of y(n) until n=p, ~~y(0)~~ x(0), …, ~~y(n-1)~~ x(n-1)are convoluted with the nth-order prediction coefficients $\alpha^{(n)}{}_1, \ldots, \alpha^{(n)}{}_n$ by the following calculation to obtain a prediction value:

$$\sum_{i=1}^{n}\alpha_i{}^{(n)}y(n-1)$$

The prediction value is added to y(n) to generate a prediction synthesis signal x(n). After n=p, as is the case with the prior art, the immediately preceding ~~n prediction error~~ p reconstructed signals ~~y(n-p)~~ x(n-p), …, ~~y(n-1)~~ x(n-1) are convoluted with the pth order prediction coefficient ~~by Eq. (3)~~ to obtain a prediction value, which is added to y(n) to obtain a prediction synthesis signal x(n). In this prediction synthesis, too, by setting the prediction coefficients to the values shown in the Fig. 19 table for the current-frame samples y(n), where n=0, …, L-1, it is possible to achieve the prediction synthesis in the current frame without extending over the preceding and succeeding frames. ~~Also in this prediction synthesis, by setting the prediction coefficients as presented in Fig. 19 for the input of the current frame samples y(n), where n=0, …, L-1, it is possible to reduce the discontinuity of the prediction synthesis signals between the preceding and current frames even if the prediction synthesis processing is carried out without extending over the preceding frame.~~

Please replace the paragraph at page 42, prenumbered lines 9-19, with the following rewritten paragraph:

In the linear prediction coefficients, an ith coefficient $\alpha^{(q)}{}_i$ of an order q takes a different value in accordance with the value of the order q. Accordingly, in Embodiment [[7]] 6 described above, it is necessary that the prediction coefficient values by which the past samples are multiplied in the multiplying parts $24_1, \ldots, 24_p$ be changed for each input of the

sample x(n) in such a manner that, for example, in Fig. 3A, the 1st-order prediction coefficient $\alpha^{(1)}_1$ is used as a prediction coefficient α_1 for the input sample x(1), the 2nd-order prediction coefficients $\alpha^{(2)}_1$, $\alpha^{(2)}_2$ (other αs being 0) are used as prediction coefficients α_1, α_2 for the input sample x(2), the 3rd-order prediction coefficients $\alpha^{(3)}_1$, $\alpha^{(3)}_2$, $\alpha^{(3)}_3$ (other αs being 0) are used as prediction coefficients α_1, α_2, α_3 for an input sample x(3).

Please replace the paragraph at page 42, prenumbered line 20, to page 43, prenumbered line 10, with the following rewritten paragraph:

On the other hand, in PARCOR coefficients an ith coefficient remains unchanged even if the value of the order q changes. That is, PARCOR coefficients $k_1, k_2, \ldots, k_p$ do not depend on the order. It is well-known that the PARCOR coefficient and the linear prediction coefficient are reversibly transformed to each other. Accordingly, it is possible to calculate the PARCOR coefficients $k_1, k_2, \ldots, k_p$ from the input sample, the 1st-order prediction coefficient $\alpha^{(1)}_1$ from the coefficient k_1, and the 2nd-order prediction coefficients $\alpha^{(2)}_1$, $\alpha^{(2)}_2$ from the coefficients k_1, k_2; thereafter, (p-1)th-order prediction coefficients $\alpha^{(p-1)}_1, \ldots, \alpha^{(p-1)}_{p-1}$ can similarly be obtained from the coefficients $k_1, \ldots, k_{p-1}$. This calculation can be expressed as follows:

For i=1: $\alpha^{(1)}_1=k_1$

For i=2, …, p; $\alpha^{(i)}_i=-k_1$

$$\alpha^{(i)}_j=\alpha^{(i-1)}_j-k_i\alpha^{(i-1)}_{i-j},\ j=1, \ldots, i-1$$

This calculation can be conducted in a shorter time and hence more effectively than in the case of calculating $\{\alpha^{(1)}_1\}$, $\{\alpha^{(2)}_1, \alpha^{(2)}_2\}$, $\{\alpha^{(3)}_1, \alpha^{(3)}_2, \alpha^{(3)}_3\}, \ldots, \{\alpha^{(p-1)}_1, \alpha^{(p-1)}_2, \ldots, \alpha^{(p-1)}_{p-1}\}$ by linear prediction for the sample number n=1, …, p-1 as described previously with reference to Embodiment <u>6 and</u> 7.

Please replace the paragraph at page 43, prenumbered lines 23-26, with the following rewritten paragraph:

Upon input of x(2), the prediction coefficient determining part 53 calculates 2nd-order prediction coefficients $\alpha^{(2)}_1$, $\alpha^{(2)}_2$ from k_1 and k_2, and sets them in the corresponding multiplier, from which is output a 2nd-order prediction error $y(2)=x(2)-[\cancel{\alpha^{(2)}_1x(0)+\alpha^{(2)}_2x(1)}\ \underline{\alpha^{(2)}_2x(0)+\alpha^{(2)}_1x(1)}]$.

Please replace the paragraph at page 43, prenumbered line 27, to page 44, prenumbered line 3, with the following rewritten paragraph:

Upon input of x(3), the prediction coefficient determining part 53 calculates 3rd-order prediction coefficients $\alpha^{(3)}_1$, $\alpha^{(3)}_2$, $\alpha^{(3)}_3$ from k_1 k_2 and k_3, and sets them in the corresponding multiplier, from which is output a 3rd-order prediction error $y(3)=x(3)-[\cancel{\alpha^{(3)}_1x(0)+\alpha^{(3)}_2x(1)+\alpha^{(3)}_3x(2)}\ \underline{\alpha^{(3)}_3x(0)+\alpha^{(3)}_2x(1)+\alpha^{(3)}_1x(2)}]$.

Please replace the paragraph at page 49, prenumbered line 21, to page 50, prenumbered line 5, with the following rewritten paragraph:

Fig. 28A illustrates an embodiment of the present invention as being applied, for example, to the FIR filtering in the ~~up-converting~~ up-sampling part 16 in Fig. 1. In the buffer 100 there are stored samples x(0), …, x(L-1)of the current frame FC. As described previously with reference to Figs. 2A, 2B and 2C, in the case of FIR filtering, a convolution is usually carried out, for the sample x(n) at each point in time n, between that sample and T preceding and succeeding samples, i.e. a total of 2T+1 samples, and coefficients $h_1, \ldots, h_{2T+1}$, but in the case of applying the present invention to the FIR filtering, no samples of the preceding frame are not used, but instead, as shown in the table of Fig. 28B, the tap number of the FIR filter is increased for each sample from the first sample x(0) to the sample x(T) in

the current frame, and after the sample x(T) filtering with a predetermined tap number is performed.

Please replace the paragraph at page 50, prenumbered lines 6-21, with the following rewritten paragraph:

Figs. 28A and 28B exemplify filtering in the case of T=2 for the sake of brevity. A prediction coefficient determining part 101 is supplied with samples x(0), x(1), ... and, based on them, calculates prediction coefficients h0, h1, ... for each sample number n as shown in the table of Fig. 28B. The sample x(0) of the current frame, read out of the buffer 100, is multiplied by a multiplier 22_0 by the coefficient h_0 to obtain an output sample y(0). Then a convolution is carried out, by multipliers 22_0, ~~22_2~~ $\underline{22_1}$, ~~22_3~~ $\underline{22_2}$ and an adder 23_1, between samples x(0), x(1), x(2) and the coefficients h_0, h_1, h_2 to obtain an output y(1). Then a convolution is carried out, by multipliers 22_0, ..., 22_4 and an adder 23_2, between samples x(0), ..., x(4) and the coefficients h_0, ..., h_4 to obtain an output y(2). Thereafter until n=L-1 is reached, a convolution is carried out between the sample x(n) and four samples preceding and succeeding it, i.e., a total of five samples and the coefficients h_0, ..., h_4 to obtain the output y(n). After this, since the number of remaining samples of the current frame is smaller than T, the tap number of filtering is decreased one by one.

Please replace the paragraph at page 55, prenumbered lines 4-9, with the following rewritten paragraph:

The sample sequence y(0), ..., y(L-1) is fed to the prediction synthesis part 63, with the first sample in the head (S1). The sample sequence is subjected to the prediction synthesis processing to generate a prediction synthesis signal ~~v(n)~~ <u>v(n)'</u> (where n=0, ..., L-1) (S2). The prediction synthesis signal v(n)' is temporarily stored in the buffer 100. This

prediction synthesis utilizes the scheme described previously with reference to Fig. 20 or 21B.

Please replace the paragraph at page 55, prenumbered lines 10-23, with the following rewritten paragraph:

In the auxiliary information decoding part 330 the auxiliary code CAI, which forms part of the code of the current frame FC, is decoded into auxiliary information, from which τ and β are obtained (S3). In some cases, the auxiliary information itself is input to the auxiliary information decoding [[320]] 330. In the sample sequence acquiring part 340 a sample sequence $v(\tau)$, ..., $v(\tau+p)$ consisting of a predetermined number p, in this example, of consecutive samples, is replicated from the synthesis signal (sample) sequence v(n) by use of τ, that is, the sample sequence $v(\tau)$, ..., $v(\tau+p)$ is acquired with the prediction synthesis signal sequence v(n) unchanged (S4), and this sample sequence is shifted to bring its forefront to the front position in the frame FC to obtain a sample sequence u(n), which is multiplied in the gain multiplying part 350 by the gain β obtained from the auxiliary information, thereby generating a corrected sample sequence $u(n)'=\beta u(n)$ (S5).

Please replace the paragraph at page 58, prenumbered line 21, to page 59, prenumbered line 12, with the following rewritten paragraph:

In the absence of the code set of the preceding (past) frame, for example, when the code set (Im, Pe, C_A) of the preceding frame is not available due to packet dropout during transmission, or when decoding is started from the code set of an intermediate one of a plurality of consecutive frames for random access, the absence of the code set of the preceding frame is detected in a dropout detecting part 450, then the auxiliary code C_A (or C_A') (the auxiliary code CA or CA' described previously with reference to Embodiment 13)

separated in the separating part 32 is decoded in an auxiliary <u>information</u> decoding part 460 into the auxiliary sample sequence x(-p), ..., x(-1) (or x(0), ..., c(p-1)), then this auxiliary sample sequence is input as a prediction-synthesis rear-end sample sequence x(-p), ..., c(-1) to the prediction synthesis part 63, then the prediction error signals y(0), ..., y(L-1) of the current frame are sequentially input to the prediction synthesis part 63, which performs prediction synthesis to generate the synthesis signal x(), ..., x(L-1). The auxiliary code C_A (C_A') is double and hence is redundant, but a prediction synthesis signal of excellent continuity and quality can be obtained. The decoding scheme in the auxiliary <u>information</u> decoding part 460 is a scheme corresponding to the coding scheme in the auxiliary information coding part 420 in Fig. 36.

Please replace the paragraph at page 59, prenumbered lines 13-22, with the following rewritten paragraph:

In the above there has been described, with reference to Figs. 36 to 39, the digital signal processing associated with, for example, the prediction error generating part 51 in the coder 10 and the prediction synthesis part 63 in the decoder in Fig. 1, but the same scheme as described above is also applicable to the digital signal processing associated with the FIR filter of Fig. 2A which is used in the ~~up-converting~~ <u>up-sampling</u> parts 16 and 34 in Fig. 1. In such a case, the prediction error generating part 51 in Fig. 36 and the prediction synthesis part 63 in Fig. 38 are each substituted with the FIR filter of Fig. 2A as indicated in the parentheses. The procedure for signal processing is exactly the same as described previously with respect to Figs. 36 to 39.

Please replace the paragraph at page 61, prenumbered line 20, to page 62, prenumbered line 10, with the following rewritten paragraph:

In this practical embodiment the backward prediction part 511 performs linear prediction backward of the header symbol of the random-access starting frame. The prediction error generating part 51 performs forward linear prediction for the samples of frames. The decision part 512 encodes the prediction error obtained by the forward linear prediction of the samples of the random-access starting frame by the prediction error generating part 51 and encodes the prediction error obtained by the backward linear prediction of the samples of the starting frame by the backward linear prediction part 511, then compares the amounts of codes, and provides select information SL for selecting the code of the smaller amount to a select part 513. The select part 513 selects and outputs the prediction error signal y(n) of the smaller amount of code for the random-access starting frame, and for the subsequent frames the select part selects the output from the prediction error generating par 51. The select information SL is coded in the auxiliary information coding part 514 and output therefrom as the auxiliary code [[CA]] $\underline{C_A}$.

Please replace the paragraph at page 64, prenumbered lines 5-21, with the following rewritten paragraph:

As depicted in Fig. 42B, the processing part 600 of the decoder 30 includes an auxiliary information decoding part 632, ~~an error code~~ <u>a</u> decoding part 640, a decoding table 641, and the prediction synthesis part 63. The auxiliary information decoding part 632 decodes the auxiliary code CA from the separating part 32, and provides the select information ST to the ~~error code~~ decoding part 640. The decoding table 641 uses the same table as the coding table 530 in the coder 10 of Fig. 42A. The ~~error code~~ decoding part 640 decodes two prediction error codes Pe for the first and second samples of the random-access

starting frame by use of the decoding table T1, and outputs the prediction error signal samples y(0) and y(1). The error code decoding part decodes the subsequent prediction error codes Pe by using the table T2 or T3 specified by the select information ST for each plurality of codes mentioned above, and outputs the prediction error signals ample y(n). The prediction synthesis part 63 performs the prediction synthesis processing described previously with reference to Fig. 20 or 21, and carries out the prediction synthesis processing of the prediction error signal y(n) and outputs the prediction synthesis signal x(n).